

Inteligência de Enxame: *MOPSO*

 Otimização multi-objetivo envolve m funções objetivo:

$$f_1, f_2, \dots, f_i, \dots, f_m$$

 A relação entre duas soluções x e y pode ser classificada em 4 categorias:

★ x domina y que é equivalente a y é dominada por x :

★ Para qualquer i , $1 \leq i \leq m$ tem-se $f_i(x)$ é melhor do que $f_i(y)$

★ x não dominada y :

★ Para pelo menos i , $1 \leq i \leq m$ tem-se $f_i(x)$ é pior do que $f_i(y)$

★ x e y são iguais:

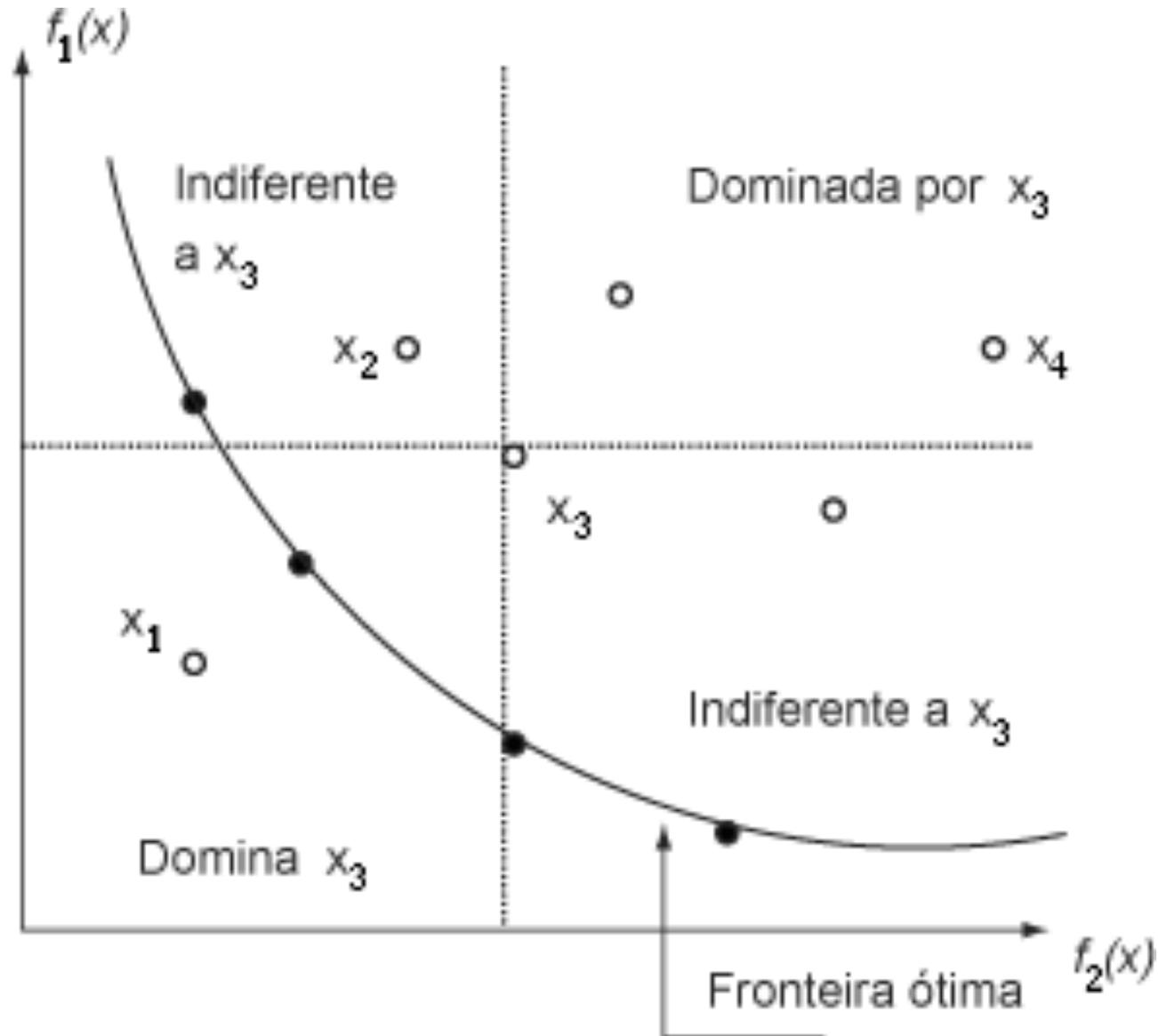
★ Para qualquer i , $1 \leq i \leq m$, tem-se $f_i(x) = f_i(y)$

★ x e y são indiferentes:

★ Existe pelo menos i e j , $1 \leq i, j \leq m$ tal que

$f_i(x)$ é melhor do que $f_i(y)$ e $f_j(y)$ é melhor do que $f_j(x)$

Inteligência de Enxame: *MOPSO*



Inteligência de Enxame: *MOPSO*

-  Inicializar o enxame de partículas com posições x_i e velocidades v_i aleatórias;
-  Avaliar as aptidões das partículas com relação a cada objetivo;
-  Inicializar p_i e $pbest_i$ para cada partícula do enxame
-  Inicializar uma tabela T como as posições das partículas não-dominadas;
-  Repita
 - ★ Calcular a distância de superlotação associada com cada posição em T ;
 - ★ Ordenar T em ordem decrecente dos valores de distância de superlotação;
 - ★ Escolher aleatoriamente, dos ξ primeiros elementos da tabela T , uma posição g e considere-a como a da melhor partícula;

Inteligência de Enxame: *MOPSO*

 ... (continuação)

★ Para cada partícula do enxame Faça

★ Calcular a nova velocidade v_i , considerando como a melhor posição da partícula p_i e a melhor posição do enxame g ;

★ Calcular a nova posição x_i ;

★ Realizar a operação de mutação;

★ Avaliar as aptidões $fitness_k$, $1 \leq k \leq m$ da partícula;

★ Se $pbest_{ik}$ melhor do que $fitness_k$ para $1 \leq k \leq m$

✦ Atualizar p_i com x_i

★ Atualizar T com base na nova posição x_i da partícula;

 Até que o critério de parada seja verdadeiro;

Inteligência de Enxame: *MOPSO*



Atualização de T com base na posição x_i

- ★ Se não existir nenhuma posição y em T | y domina x_i
 - ★ Remover todas as posições em T que são dominadas por x_i ;
 - ★ Se T estiver cheia
 - ★ Ordenar T em ordem crescente dos valores de distância de superlotação;
 - ★ Escolher aleatoriamente, dos ξ primeiros elementos da tabela T , uma posição e a substituir com x_i ;
 - ★ Senão inserir x_i em T mantendo a ordem;
- ★ Senão descartar x_i

Inteligência de Enxame: *MOPSO*

-  A distância de superlotação de uma posição x fornece uma estimativa da densidade de posições de partículas em torno da x ;
-  O guia global g é escolhido em regiões menos lotadas;
-  A remoção de uma posição de T é guiada pela região mais lotadas;
-  Calculo da distância de superlotação para cada posição em T
 - ★ Seja $max+1$ o número total de entradas em T ;
 - ★ Inicializar $DS[j] := 0$ para $1 \leq j \leq max-1$
 - ★ Inicializar $DS[0] = DS[|T|] := \infty$
 - ★ Para cada função objetivo f_k , $1 \leq k \leq m$ Faça
 - ★ Ordenar as posições em T em ordem decrescente do objetivo f_k ;
 - ★ Para j de 1 a $|T|$ Faça
 - ◆ $DS[j] := DS[j] + (DS[j+1] - DS[j-1])$

Inteligência de Enxame: *MOPSO*

Operação de mutação de partículas

Algoritmo de mutação do MOPSO

Se $rand(0, 1) < taxa$ Então

$dim := rand(0, d - 1)$

$inter := rand(max_{dim}, min_{dim})$

$nmax := max_{dim} - inter$

$nmin := min_{dim} + inter$

Se $nmin < min_{dim}$ Então

$nmin = min_{dim}$

Se $nmax > max_{dim}$ Então

$nmax = max_{dim}$

$x_{dim} := rand(nmin, nmax)$

Fim Se

Fim mutação do MOPSO

Inteligência de Enxame: *MOPSO*

 O critério usado para eleger a melhor posição da partícula i leva em conta $pbest_{ik}$ e $fitness_k$ para cada função objetivo f_k , $1 \leq k \leq m$.

★ Se p_i é dominada por x_i Então

★ Eleger x_i

★ Se x_i é dominada por p_i Então

★ Eleger p_i

★ Senão

★ Eleger p_i ou x_i de acordo com uma amostragem aleatória